REMARKS

Reconsideration of the rejections set forth in the Office Action is respectfully requested. By this amendment, claims 2-3 and 5 have been cancelled without prejudice or disclaimer and claims 1, 4, and 6-7 have been amended. Currently, claims 1, 4, 6-10 and 15-22 are pending in this application.

Rejection under 35 USC 101

Claim 10 was rejected under 35 USC 101 as being directed to non-statutory subject matter. Specifically, the Examiner has taken the position that a protocol data unit is a data structure, and that a data structure per se is non-statutory. According to the Examiner, merely putting a data structure on a computer readable medium does not make it statutory because a protocol data unit on a computer readable medium is not capable of causing any functional change in the computer and, thus, does not produce a useful result. As basis for this, the Examiner has cited MPEP 2106.01. For convenience, the beginning portion of MPEP 2106.01 is reproduced below:

> Descriptive material can be characterized as either "functional descriptive material" or "nonfunctional descriptive material." In this context, "functional descriptive material" consists of data structures and computer programs which impart functionality when employed as a computer component. (The definition of "data structure" is "a physical or logical relationship among data elements, designed to support specific data manipulation functions." The New IEEE Standard Dictionary of Electrical and Electronics Terms 308 (5th ed. 1993).) "Nonfunctional descriptive material" includes but is not limited to music, literary works, and a compilation or mere arrangement of data.

> Both types of "descriptive material" are nonstatutory when claimed as descriptive material *per se*, 33 F.3d at 1360, 31 USPQ2d at 1759. When functional descriptive material is recorded on some computer-readable medium, it becomes structurally and functionally interrelated to the medium and will be statutory in most cases since use of technology permits the function of the descriptive material to be realized. Compare *In re Lowry*, 32 F.3d 1579, 1583-84, 32 USPQ2d 1031, 1035 (Fed. Cir. 1994)(discussing patentable weight of data structure limitations in the context of a statutory claim to a data structure stored on a computer readable medium that increases computer efficiency) and >In re<

> *Warmerdam*, 33 F.3d *>1354,< 1360-61, 31 USPQ2d *>1754,< 1759 (claim to computer having a specific data structure stored in memory held statutory product-by-process claim) with *Warmerdam*, 33 F.3d at 1361, 31 USPQ2d at 1760 (claim to a data structure *per se* held nonstatutory).

There are two clear statements in this section:

(1) "functional descriptive material" consists of data structures and computer programs which impart functionality when employed as a computer component; and

(2) When functional descriptive material is recorded on some computer-readable medium, it becomes structurally and functionally interrelated to the medium and will be statutory in most cases since use of technology permits the function of the descriptive material to be realized

Indeed, the fact that data structures are patentable is so well ingrained in US Patent law that the Patent Office has a classification for such data structures (U.S. Patent Class 707/100). Thus, the Examiner's statement that "A data structure is per-se non-statutory" is inconsistent with the MPEP and incorrect. Accordingly, the Examiner is respectfully requested to withdraw the rejection.

The second part of the Examiner's logic as to why claim 10 is not statutory is that "a protocol data unit on a computer readable medium is not capable of causing any functional change in the computer, [and] thus does not produce a useful result." This statement is unsupported by any factual basis.

Claim 10 recites that the protocol data unit data structure stored in a tangible computer readable medium includes a destination MAC address that has a plurality of fields, each of which contains "a code to be used by a switch on a network to identify an output port on the switch..." When the switch uses the code from the data structure to "identify an output port on the switch" the data structure causes a function change within the switch. Namely, it causes the switch to determine an output port. That is, in essence, what a switch does – it switches data. Thus, the data structure clearly causes a functional change because it causes the switch to operate to identify an output port.

Applicants respectfully submit that (1) data structures when recorded on a computer readable medium are presumptively patentable (Note that the MPEP states that functional

descriptive material is patentable in "most cases"); and (2) applicants have established that claim 10 also recites a particular functional change / useful result that is caused in the computer by the data structure. Accordingly, the Examiner is respectfully requested to withdraw the rejection under 35 USC 101.

Rejection under 35 USC 102

Claims 15, 16, and 19 were rejected under 35 USC 103 as anticipated by Pearce et al (U.S. Patent No. 6,556,574). This rejection is respectfully traversed in view of the amendments to the claims and the following arguments.

Claim 15 recites a method of assigning a MAC address to an interface on a network. The first step in claim 1 recites setting a local bit in the MAC address to indicate to network elements on the network that the MAC address is locally assigned. Pearce shows a frame format of a token ring. The token ring frame format includes a 48 bit destination address. (Pearce at Col. 14, lines 5-6 "Destination addresses always consist of six 8-bit bytes."). The second bit (identified by reference numeral 606 of byte 0 is used in the Token Ring standard (802.5) to designate whether the bit is global or local. (Pearce at Col. 14, lines 10-14). In 802.5 parlance, this is referred to as the U/L bit, which indicates whether the address is Universal or Locally administered. Thus, Pearce clearly shows this first step of the method of claim 15. This is not surprising, however, as applicants explain in paragraph 24 of the specification that the IEEE defined MAC address format includes a bit that is reserved to indicate whether the MAC address is a global MAC address or is a local MAC address. Applicants are prepared to admit that local MAC addresses were known at the time the application was filed.

The Examiner has taken the position that Pearce teaches "assigning a first value to a first field of the MAC address, the first field containing a smaller number of bits than a total number of bits of the destination MAC address". As support for this, the Examiner has cited field 604 as containing less number of bits than the total number of bits in the MAC address. (See Office Action at page 3). In Pearce, reference numeral 604 is used to refer to Byte 0 of the destination address. As noted by Pearce in Col. 14, lines 5-6, destination addresses "always consist of six 8-bit bytes." Further, Pearce continues to explain that "Bit "0" 602 of byte "0"604 is called the I/G bit…" Thus, what the Examiner has cited as a first field is the first Byte of a standard token ring physical address.

The Examiner then contends that Pearce teaches that "the first value containing first output interface information usable by a first switch to identify a first output interface for transmission of frames containing the first value in the first filed of the MAC address." As support for this, the Examiner has cited col. 20, lines 10-18.

The portion of Pearce cited by the Examiner refers to MAC addresses as a whole. Nothing in this portion suggests that a switch would read only the first byte (field 604) and use the value of that first byte to determine an output port.

Claim 15 recites

(1) a first value is assigned to a first field of the MAC address, and that the first field has a smaller number of bits than the total number of bits of the destination MAC address; and

(2) that the first value contains first output interface information usable by a first switch to identify a first output interface.

Pearce teaches a token ring switch that forwards frames based on standard 48 bit physical addresses. In an attempt to read Pearce onto this claim, the Examiner has focused on the fact that the 48 bit physical address if made up of 6- 8bit bytes. The fact that a token ring address has a first byte does not mean that the value contained within the first byte is usable by the first switch to identify an output interface. Hence, the fact that each address has a first byte does not cause claim 15 to be anticipated. Rather, the Examiner must show the second part of this method step, which is that the information contained within this field is used, by itself, to identify an output port. Pearce does not do this, and the Examiner has not shown that Pearce provides this teaching. Accordingly, the rejection of claims 15-17 over Pearce should be withdrawn.

One aspect that bears further explanation is table 10. The Examiner cited Col. 20, lines 10-20, which refer to Fig. 10. Fig. 10 is an ARP table (Pearce at col. 19, lines 66-67). This ARP table provides the RIF (route information) along with the MAC address, and also gives a binding between the Layer 2 (MAC layer) and Layer 3 (IP layer) addresses. (Pearce at col. 20, lines 1-9). This does not teach or suggest that the first byte of one of the MAC addresses should be usable by a switch to determine an output interface. Rather, Table 10 is an ARP table that enables the full MAC address to be combined with RIF information, and enables the MAC address to be correlated to the IP address. Thus, Table 10 does not provide support for the Examiner's position. In view of this further explanation, applicants respectfully submit that Pearce fails to

teach or suggest each limitation of independent claim 15. Accordingly, the Examiner is respectfully requested to withdraw the rejection of claim 15.

Rejection under 35 USC 102 and 35 USC 103

Claims 1, 3, 5, 6, and 10 were rejected under 35 USC 103 as unpatentable over Schaub (U.S. Patent No. 7,190,695) in view of Sandstrom (U.S. Patent No. 7,254,138). Claim 2 was rejected over Schaub and Sandstrom, and further in view of Hughes (U.S. Patent No. 7,277,399). This rejection is respectfully traversed in view of the amendments to the claims and the following arguments.

Applicants have amended independent claim 1 to recite that the method includes the steps of extracting frame contained destination information from a Media Access Control (MAC) address associated with the received frame by reading a field within the MAC address, the field within the MAC address being a selected number of bits of the MAC address smaller than the total number of bits of the MAC address and located at a particular location within the MAC address; and making a switching decision within the first switch based on the extracted frame contained destination information without performing a lookup in a forwarding table to determine an output port from the first switch over which the frame should be forwarded onto the communication network.

Several of these limitations that have been added to independent claim 1 were formerly set forth in dependent claims 2, 3, and 5, which have now been cancelled. In connection with these claims, the Examiner has taken the position that Schaub teaches reading a field of a MAC address that is smaller than the entire MAC address. (Office Action at page 5). Further, the Examiner has taken the position that Schaub teaches using the information that is read from the field to identify the output port. As support for these positions, the Examiner has cited Schaub at Col. 7, lines 56-60, and Col. 8, line 59 to Col. 9, line 4.

For convenience, Col. 7, lines 56-60 of Schaub is reproduced below:

The parsed fields for each packet are forwarded to the categorizer unit 542 through example connections 554. The categorizer unit reads some or all of the parsed fields of each packet to determine the category in which the packet belongs.

This cited passage certainly does use the word "field," however it does not teach or suggest that the field should be a portion of the MAC address that is contained within the MAC address and smaller than the entire MAC address. Rather, the word "field" is used by Schaub generically to refer to a particular address or other such logical entity within a header of a packet.

To put this in context, Schaub teaches immediately before this cited passage that "At the parser, fields of interest from the header of each packet are parsed from the packet. In an embodiment, fields of interest from a packet header include the layer 2 (i.e., MAC) source address, the layer 2 (i.e., MAC) destination address…" (Schaub at Col. 7, lines 46-51). As is clear from this description, the parser is looking at the entire MAC source address or entire MAC destination address. Each of these addresses is a "field" in Schaub. Accordingly, the cited portion of Schaub does not support the Examiner's position that Schaub teaches reading a portion of a MAC address and making a forwarding decision based on that portion.

At Col. 8, lines 59-66, Schaub teaches that a hashing unit takes the parsed fields, i.e. the destination MAC address and the Source MAC address, and applies hashing algorithms to the parsed fields. The fields referred to in this portion of Schaub are the same fields that are discussed at Col. 7, lines 46-55, namely the destination MAC address, source MAC address, MPLS label, VLAN ID, Ethernet type/length field, IP address, etc. (See also Schaub at Col. 9, lines 1-4, stating that the hashing function operates on the layer 2 source and destination address fields.) Thus, although Schaub uses the same term "field" as applicants do, he has a different meaning for this term and uses the term to refer to the entire MAC address, not a sub-field within the MAC address.

As noted previously in earlier amendments, this application relates to a way to allow a switch to directly identify the output port for a given frame without performing a table access operation. (Specification at Paragraph 18). One of the ways that applicants proposed to do this, was to use the portion of the header normally used as the destination MAC address to contain source routing information that would specify a particular path through the switches on the local network. In this embodiment, rather than look at the whole destination MAC address (DA) and perform a FIB lookup based on the DA, the switches would look at a particular field within the DA and switch based on that content of that field.

Schaub addresses how packets in an incoming packet stream may be distributed to multiple parallel slower streams for processing within a network element. (see Schaub at Col. 1,

line 62 to Col. 2, line 6). Schaub explains that there are different types of packet flows, some of which contain packets that may be processed out of order and others of which contain packets that must be processed in-order. (Schaub at Col. 2, lines 10-17). Schaub proposes to classify packets prior to distributing the packets, so that "in-order" packets are sent to the same lower bandwidth link for processing within the network element while other packets are distributed among the lower bandwidth links in a round-robin fashion to perform load balancing between the available links. (Schaub at Col. 3, lines 33-41).

Schaub does not teach or suggest using sub-fields within the MAC address to distribute packets for processing within the network element. Rather, Schaub parses the headers to extract fields from the headers such as the destination MAC address or source MAC address. Depending on the type of packet being handled, other fields may be used as well, such as the MPLS label or IP address. Schaub then performs a hashing function on this address to assign the packet/frame to one of the internal links. Performing a hashing function is generally faster than performing a table lookup, so this enables Schaub to quickly classify the packet/frame to enable flows of packets/frames to be distributed to the same lower bandwidth processing link for processing. Indeed, Schaub is presumably not able to perform table lookups for the packets/frames on the higher bandwidth link – if Schaub was able to perform table lookups for each of the packets/frames on the higher bandwidth link it would not be necessary to distribute the packets/frames to the lower bandwidth links for processing. Since hashing functions may be implemented in hardware Schaub uses this as a way to distribute packets between the several available lower bandwidth links where the processing, such as address lookups, are performed.

The Examiner stated on page 4 of the Office Action that Schaub does not teach making a switching decision without performing a lookup in a forwarding table. However, the Examiner indicated that this was taught by Sandstrom, citing the abstract and Col. 9, lines 44-51 and col. 10, lines 12-19.

Sandstrom teaches that a forwarding instruction tag (FIT) 30 should be added to a packet to enable the packet to be transported across a lower layer (e.g. layer 2) network. The FIT includes a bit vector (31a-d) each of which identifies a particular logical connection through the Layer 2 network. (Sandstrom at Figs. 3 & 4; Col. 6 lines 15-17).

At col. 9, lines 16-31, Sandstrom explains that the bit vector can have any number of bits, and that each one of its bits is an explicit and individualized indication of whether the network

domain should deliver the packet to the next hop layer N destination with a corresponding position within the next hop destination presentation row. Thus, each bit of the bit vector (e.g. 31a, 31b, 31c, etc.) corresponds to a particular path (6a, 6b, 6c, etc.) to reach a particular destination node (upper layer node 2). Each lower layer node will forward the packet/frame on any path that has an appropriate bit set in the bit vector. (Sandstrom at Col. 6, lines 17-25).

Since the lower layer nodes do not know, in advance, where they will forward the packet/frame, each lower layer node will read the entire bit vector to determine which bits of the bit vector have been set. This is required to enable the node to forward the packet to any path that has an appropriate bit set in the bit vector – the node cannot determine whether a bit in the bit vector has been set until it reads the entire bit vector.

Thus, in sum, Sandstrom teaches that a new header should be applied to a packet which includes a Forwarding enabling vector (FEV) 30 which is a bit vector. Each bit of the bit vector identifies a particular path through the network. The network node will read the entire bit vector and use the values of the bit vector to forward the frame across the network.

Sandstrom thus does not propose that a network node should read a portion of an address field, i.e. a sub-field within a MAC address, and make a forwarding decision based on that sub-field. Rather, Sandstrom proposes to use a new header with a different format. The header includes an address field in the form of a bit vector. Rather than read a portion of the address field, the nodes on the network read the entire address field to determine how to forward the packet. Although each bit of the address field represents a different destination, the node still reads the entire address field and does not read a portion of the address field to make a forwarding decision.

Thus, applicants respectfully submit that Sandstrom would not have taught or suggested that a MAC address could be divided into fields which would have independent significance to the nodes on the network. Accordingly, applicants respectfully submit that claim 1, as amended, is patentable over the combination of Schaub and Sandstrom.

In connection with rejecting claim 2, the Examiner cited Hughes as teaching reading a portion of a header of the frame and causing the frame to be passed directly to the output port without performing a table lookup operation. As support for this the Examiner cited Hughes, element 410. Element 410 of Fig. 4 in Hughes recites "locate routing information in hardware-based route cache."

At Col. 4, line 60 to Col. 5, line 3, Hughes explains that, when a router receives a data packet, it will extract the network destination address and then perform a lookup using a hardware based cache lookup. To reduce the number of bits used to perform the lookup, the destination address may be hashed prior to being fed into the cache. This does not mean that a portion of the destination address is being used, or that a field within the destination address is used to perform a lookup. Rather, the entire destination address is read, passed through a hash function, and then the output of the hash function is used to perform the lookup. The output of the hash function will depend on the entire value of the destination address not simply on the value of a portion of the destination address. Hence Hughes, like the other cited references, is fundamentally different and does not teach or suggest that a portion of the MAC address, i.e. a field within the MAC address, should be read and used to make a forwarding decision.

Applicants have amended independent claim 1 in an attempt to clarify how forwarding is being implemented. As explained above, all three of the cited references read the entire MAC address or other address field in connection with making a forwarding decision. Schaub reads the entire MAC address and performs a hash function to distribute the packet internally to one of the lower speed links for processing. Sandstrom applies a new header with an address field (in the form of a bit vector) and then has the nodes read this entire address field to make a forwarding decision. Hughes reads the entire MAC address and performs a hash function on the MAC address to perform a lookup in a hardware route cache. Accordingly, none of the references cited by the Examiner, alone or in combination, teach or suggest the method recited in amended claim 1. Thus, applicants respectfully request that the rejection of claim 1 be withdrawn.

Conclusion

Applicants respectfully submit that this application is in condition for allowance and an action to this effect is respectfully requested. If there are any questions or concerns regarding the amendments or these remarks, the Examiner is requested to telephone the undersigned at the telephone number listed below.

<u>Extension of time</u>

Applicants request a two month extension of time to respond to the outstanding Office Action. Payment of the fee for the two month extension of time is being submitted herewith. If any additional fees are due in connection with this filing, the Commissioner is hereby authorized to charge payment of the fees associated with this communication or credit any overpayment to Deposit Account No. 141315 (Ref: 14715ROUS03U).

Respectfully Submitted

Dated: April 7, 2009

Anderson Gorecki & Manaras LLP
P.O. Box 553
Carlisle, MA 01741
Tel: (978) 264-4001
Fax: (978) 264-9119
john@gorecki.us

   /John C. Gorecki/   .
John C. Gorecki, Reg. No. 38,471